HMI Draw

User Manual.

Version 2.2

What is HMI Draw.

The HMI Draw app is the developer component of the HMI Pad system for creating Human Machine Interfaces for real time monitoring of industrial PLC based systems and processes. The other two components are the HMI View app and the HMI Pad Service.

Main features.

- ✓ Very fast native app, launches and connects immediately regardless of project size, not a web based app.
- ✓ All PLC data types supported including Boolean, Integer, Real, Strings and arrays of them.
- ✓ Advanced Expressions Engine supporting a number of data types including modern Strings, Arrays and Dictionaries.
- ✓ Projects can be fully edited on-screen as you run and monitor your process. Or can be exported and edited as a text file.
- ✓ Configurable User Accounts allow for project storage in the cloud and easy deployment to end users.
- ✓ The app connects to PLCs directly using native communication protocols. Connections are performed without any intermediate servers or boxes.
- ✓ TCP/IP based security.

The concept behind the HMI Pad apps.

The HMI Draw and HMI View apps are built on top of two main modules: the *Communications Module* and the *Expressions Engine*. These modules interact with each other and to the user interface to provide most of the underlying capabilities and many advanced features.

The **HMI Draw** lets Integrators build fully customizable HMI interfaces by adding visual items or other objects to pages. Objects have properties that can be connected together or with PLC tags through expressions. Virtually all object properties can be dynamically linked through expressions. This architecture provides an extremely powerful environment for Integrators to create advanced HMI interfaces.

HMI Interfaces can be fully created and deployed from the app by dragging and connecting items on screen. Integrators can also chose to edit project files on a text editor. The app also fully supports copy/paste/duplicate of items, tags, connections etc, and has unlimited undo/redo capabilities.

The app comes with a free service on the cloud, the **HMI Pad Service**, to store projects and associated assets for convenience and for further deployment to End Users.

The complementary **HMI View** app lets automation integrators deploy projects on End Users devices with no physical access to them. Projects on the HMiPad View app are deployed as encrypted, non-editable instances of your projects, thus helping you to keep your work and know how at safe place.

TABLE OF CONTENTS

1 The HMI Pad System Components 4

2 The HMI Draw app main user Interface 5

2.1 The Application Panel 5

2.2 The Project Viewer 6

2.3 How to: Step By Step Guide 7

2.3.1 Create a new user account 8

2.3.2 Create a new project 10

2.3.3 Edit your project 11

2.3.4 Create a PLC connection 12

2.3.5 Add tags to your PLC connector 14

2.3.6 Use an Indicator object to Read a PLC tag value 15

2.3.7 Use a Control object to Read and Write PLC tag values 16

2.3.8 Write to PLC tags without involving reads 18

2.3.9 Add custom Texts, Images and Color. 19

Document Revision History 20

1 The HMI Pad System Components

The HMI Pad System is made up of 3 components:

HMI Draw.

This is the app Integrators use to create and deploy HMI projects.

The "***HMI Draw Reference Manual***" contains concise, detailed information of every single aspect of HMI Draw development including advanced techniques and features.

HMI Pad Service.

The HMI Pad Service in the cloud works with the HMI Draw and HMI View app components.

The apps seamlessly integrate this service to provide storage options in the cloud and convenient distribution and deployment of HMI projects developed by automation integrators to end users or customers.

Refer to the "***HMI Pad Deployment Guide***" for more information on what options you have available to work with the HMI Pad Service as you develop HMI projects.

HMI View.

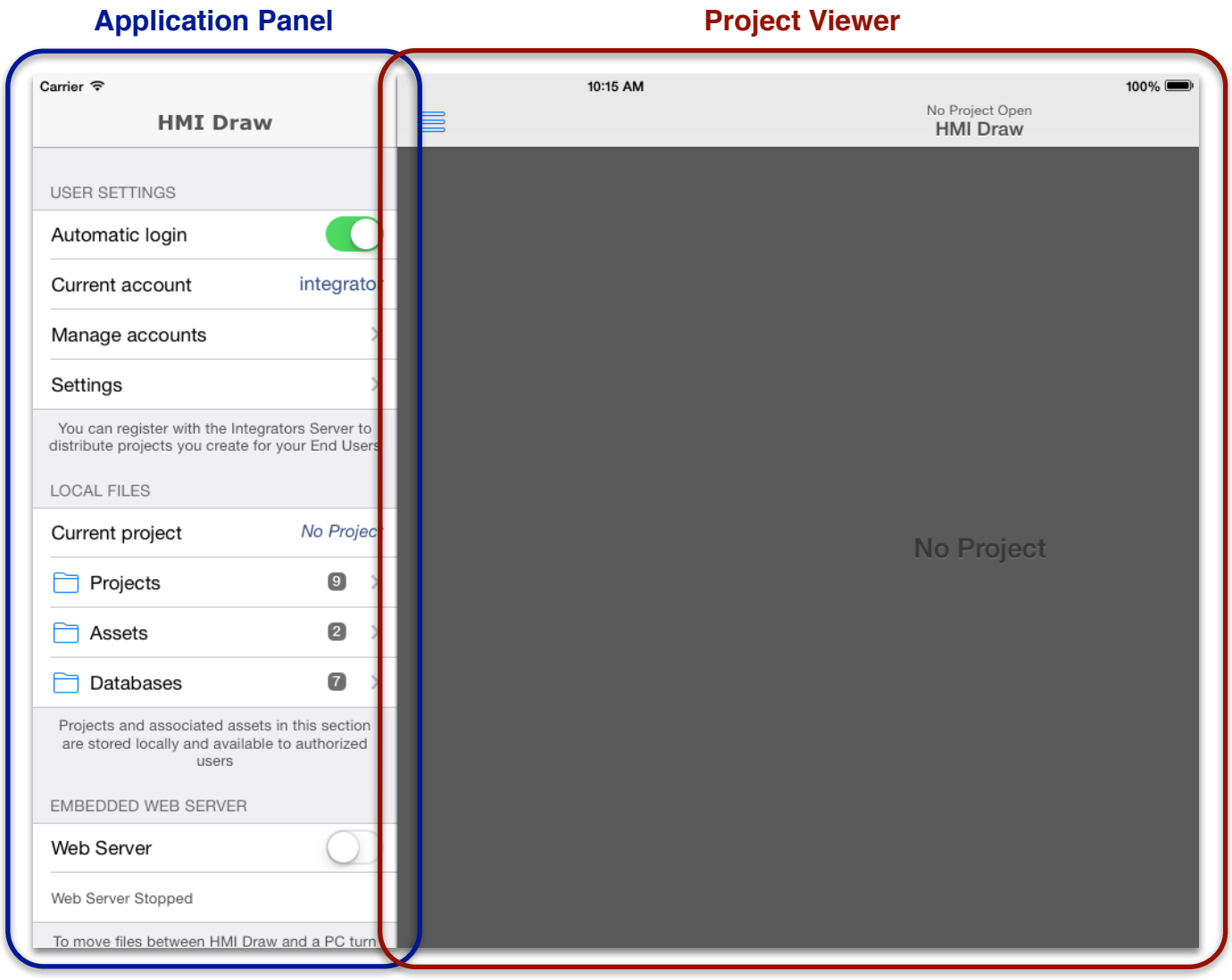
This is the app where end users run HMI projects developed by integrators. Projects on the HMI View app are installed through the HMI Pad Service and are stored as encrypted instances that can not be edited or run on unauthorized devices.

2 The HMI Draw app main user Interface

The HMI Draw app interface consists of two main draggable panels, the *Application Panel* and the *Project Viewer.*

The Application Panel is shown on the left and there you will find options and settings related with the app.

The Project Viewer is shown full screen and lets you edit, review and run your project. The Project Viewer can be dragged right to reveal the Application panel underneath.



2.1 The Application Panel

The Application Panel lets you create and manage the following aspects of the app

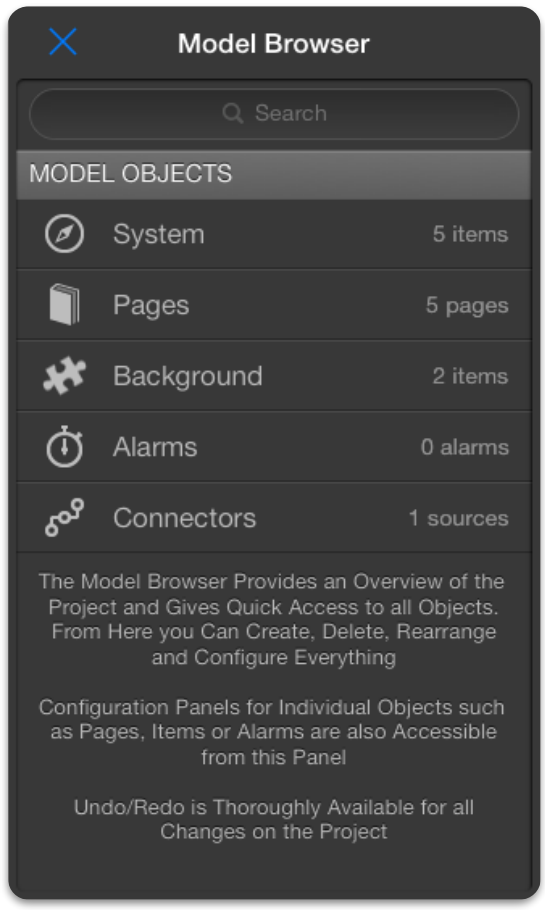
- User Accounts and Settings.
- Projects and Assets stored locally.
- Projects and Assets stored on the HMI Pad Service
- Remote Deployment.

2.2 The Project Viewer

The Project Viewer is where you create, configure, edit and run projects. From the Project Viewer you get access to Pages, Connections, Alarms and the internal aspects of your project. You can also set some editing properties and edit objects on screen using common gestures.

Several sub-panels provide means to navigate through your project to obtain detailed information. You may find the following panels that will appear at appropriate times or upon particular actions as you edit your project:

- **Page Viewer**, shows the current page. The **Page Navigator** appears from the left and displays the list of pages.
- **Inspector Panel**, appears from the right and contains the Connections Panel, the **Tag Viewer** and the **Alarms Viewer**.
- **Model Browser**, appears as a floating window you can drag around the screen.
- **Object Configurator**, appears as a floating window when you tap on 'configure' for an object.
- **Expressions Keyboard**, appears on the bottom of the screen, when editing Object Properties.
- **PLCs and TAGs Configuration Panels**, appear as floating windows, accessible from the Model Browser
- **Model Seeker,** appears as a floating window, accessible from the Object Configurators or the Expressions Keyboard


Model Browser
Search
MODEL OBJECTS
System
5 items
Pages
5 pages
Background
2 items
Alarms
0 alarms
Connectors
1 sources
The Model Browser Provides an Overview of the Project and Gives Quick Access to all Objects. From Here you Can Create, Delete, Rearrange and Configure Everything
Configuration Panels for Individual Objects such as Pages, Items or Alarms are also Accessible from this Panel
Undo/Redo is Thoroughly Available for all Changes on the Project

2.3 How to: Step By Step Guide

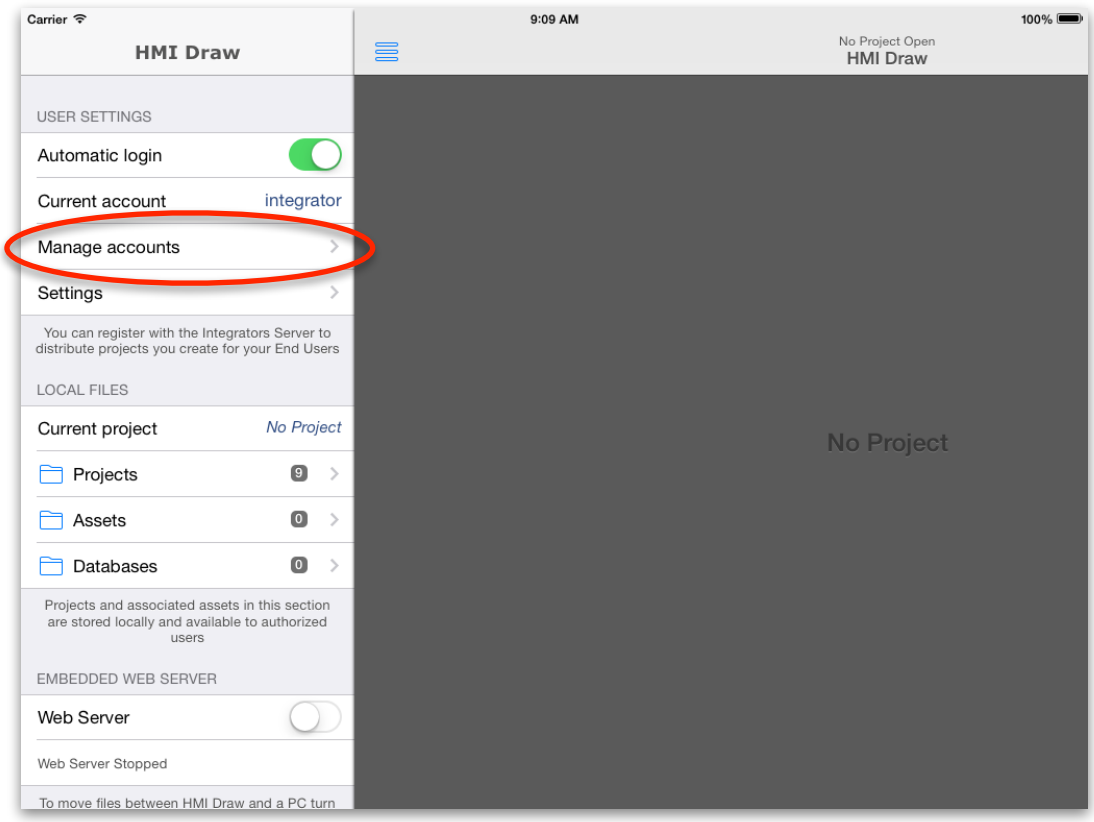
On the next sections we are presenting step by step procedures to perform essential operations.

This guide is to be considered as a starting point for HMI development using the HMI Draw app. For a complete reference of everything that is available on the app including advanced techniques and features, you can refer to the **HMI Draw Reference Manual** on a separate document.

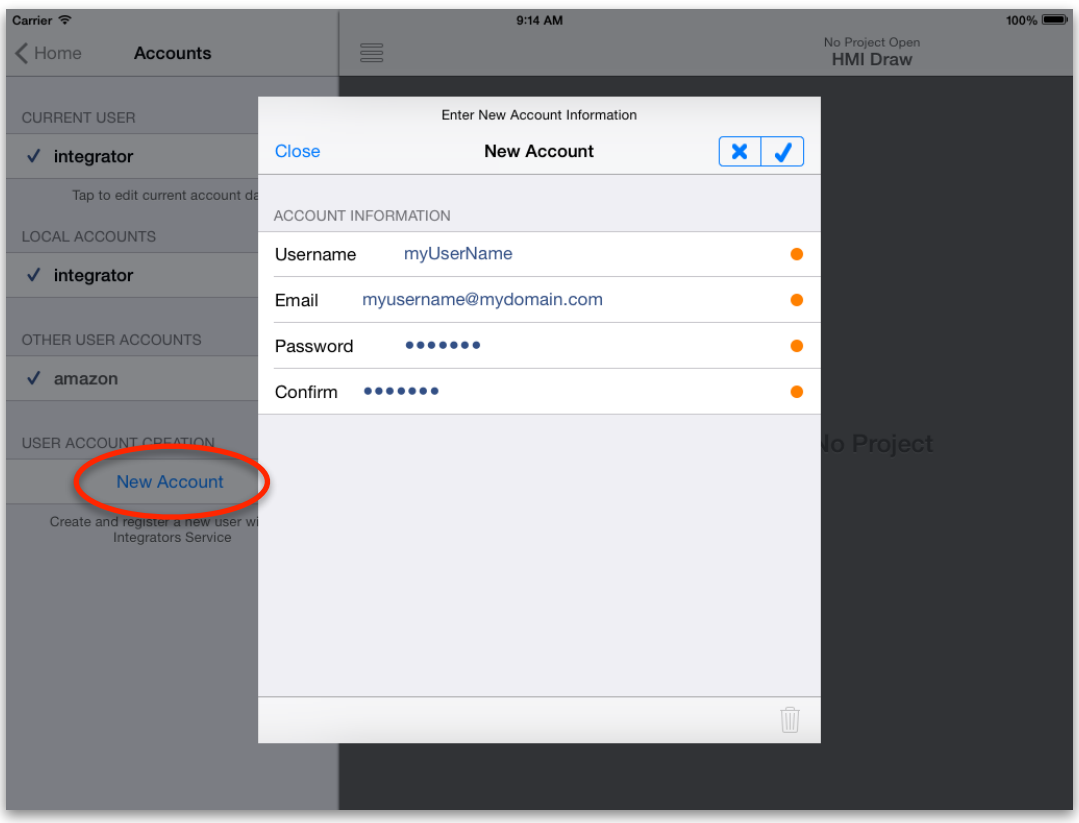
2.3.1 Create a new user account

On first launch the app logs automatically with the default user name ‘integrator’ and password ‘admin’. We recommended to create your own user account before starting using the app. User accounts let you store projects in the cloud and give unrestricted access to the HMI Pad service.

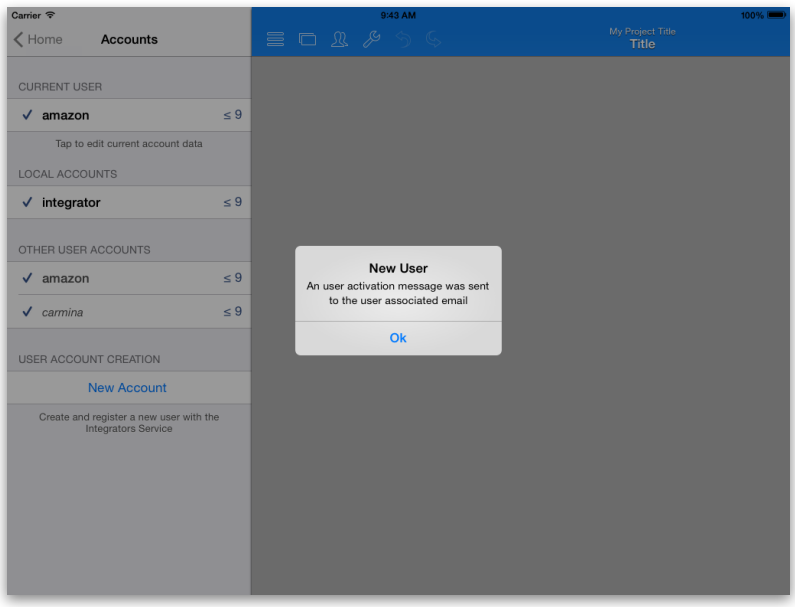
To create a new user go to the *Application Panel* and start by tapping on ‘Manage accounts’.

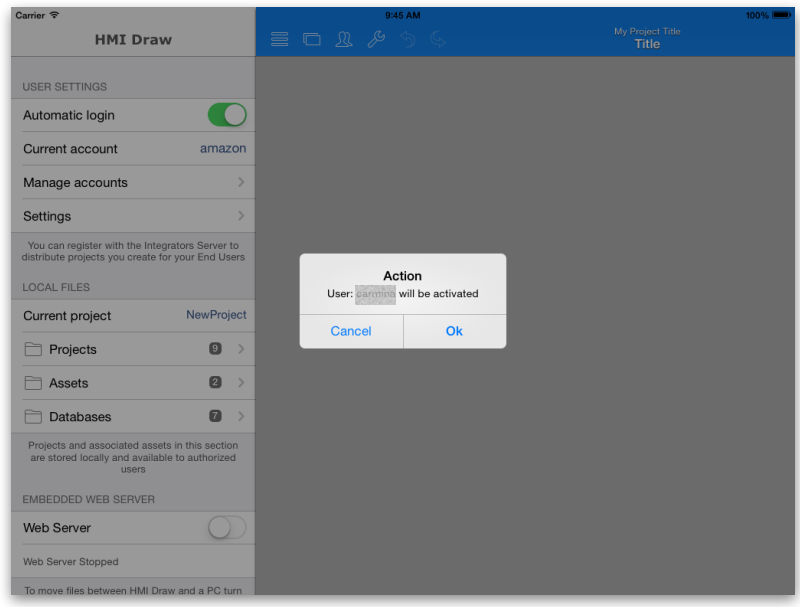


Tap on ‘New Account’ to create a new user account. Complete the form that will appear by writing a username, a valid e-mail address and a password. Once you completed the form, tap on the tick icon on the top right.

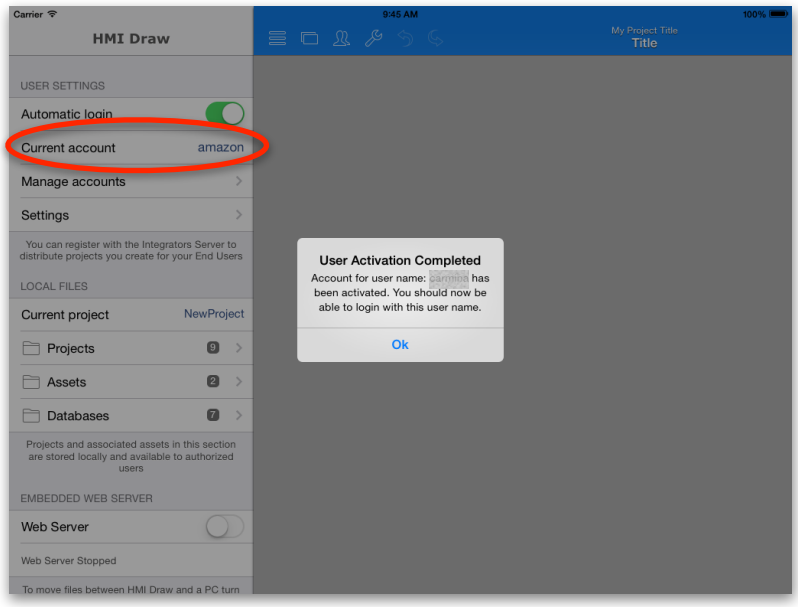


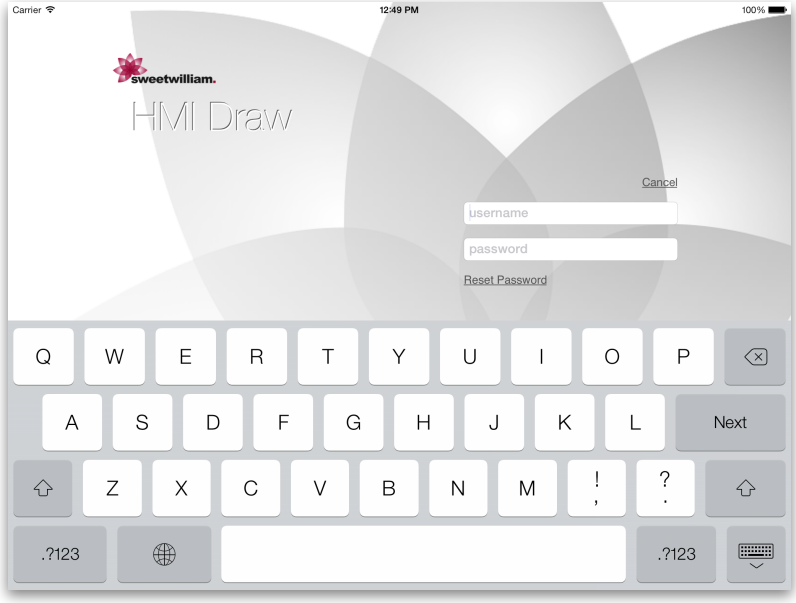
You will receive an e-mail with a confirmation link to activate your user. Open the **Mail** application on your iPad or iPhone and tap on the provided link. Alternatively you can copy the link on the **Safari** app on your iPad and press enter. The HMI Draw app will open automatically and will activate your user.





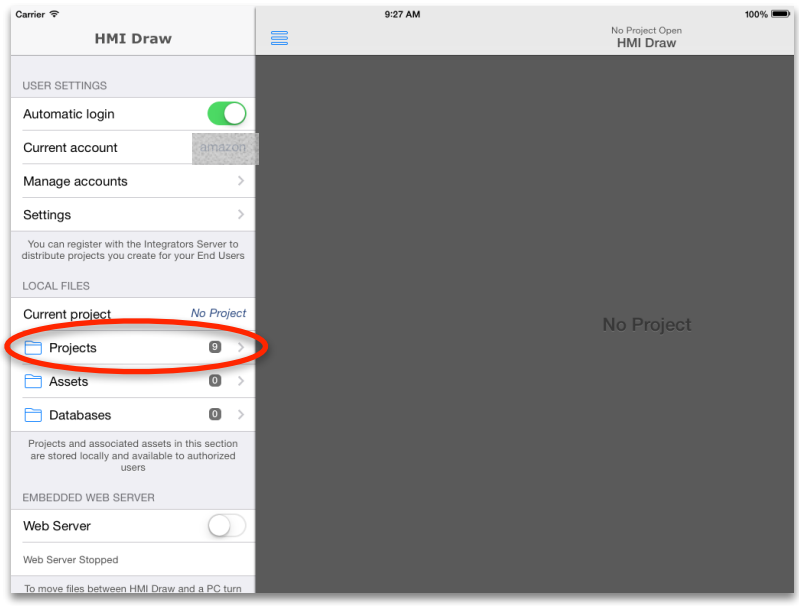
After your user activation is completed, you will be able to login with your username and password. Click on ‘Current account’ and enter your credentials (username and password). Now you are working with your own user account.



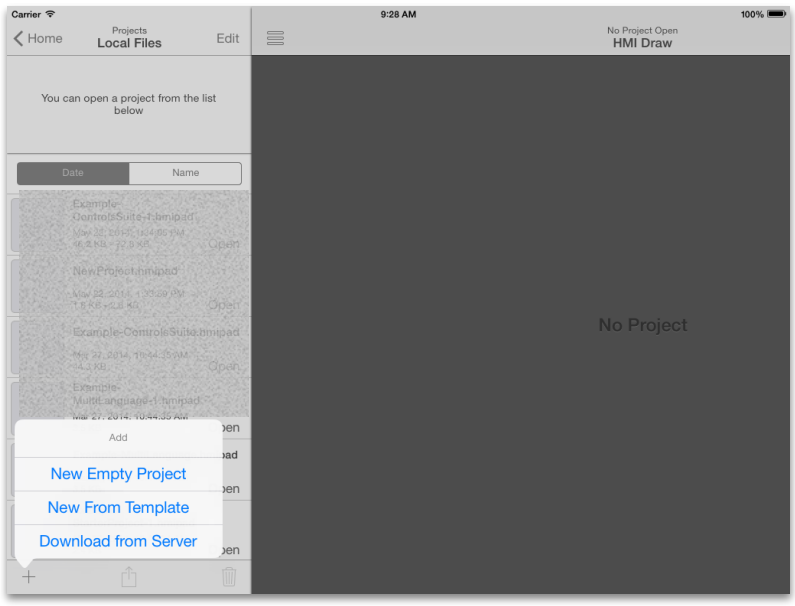


2.3.2 Create a new project

To create a new project, click on ‘Projects’ on the *Application Panel*. Then click on Add (plus sign)

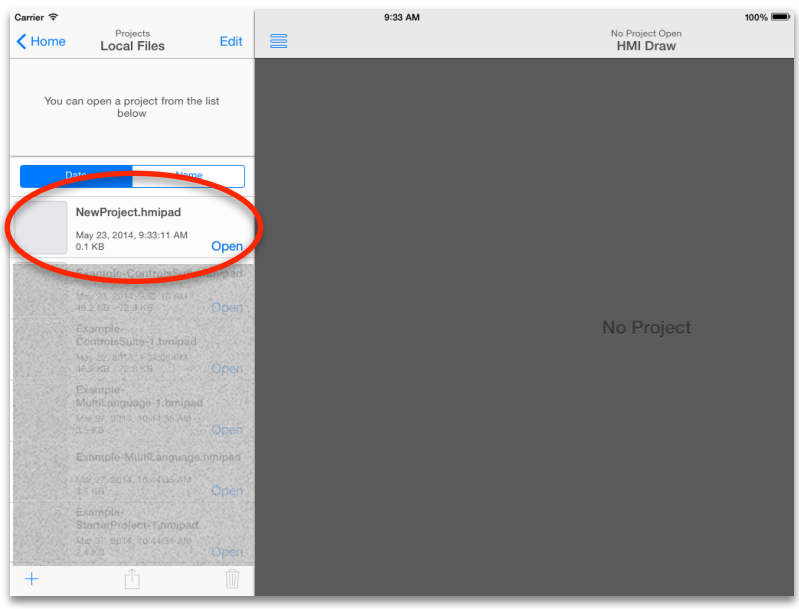


You can chose to create a ‘New Empty Project’, a ‘New project from Template’ or to download a project from a http or ftp Server.



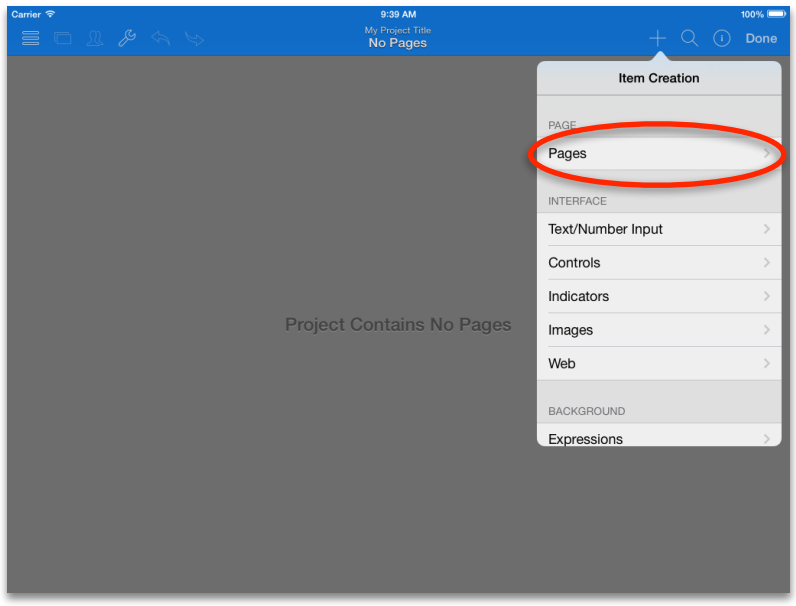
In case a ‘New Empty project’ is selected, a project named ‘NewProject.hmipad’ will be added to your local project files.

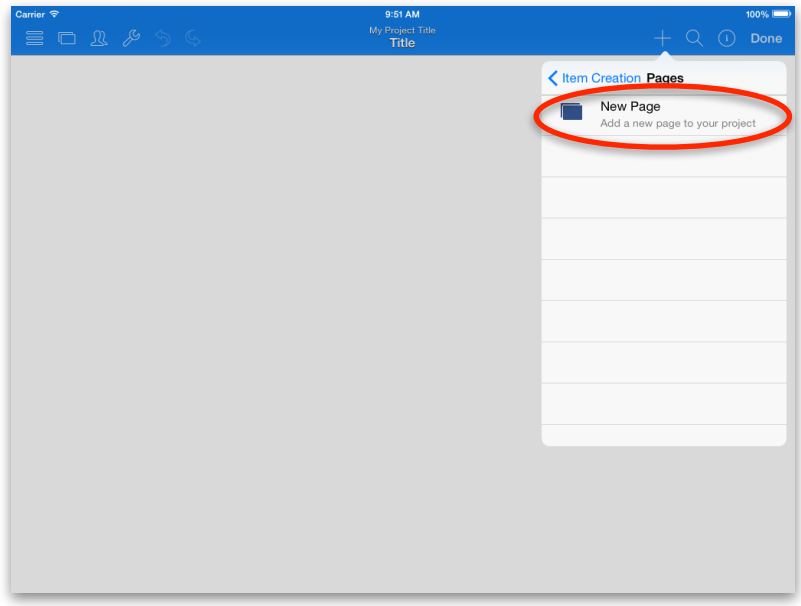
Click ‘Open’ to start developing your newly created project.



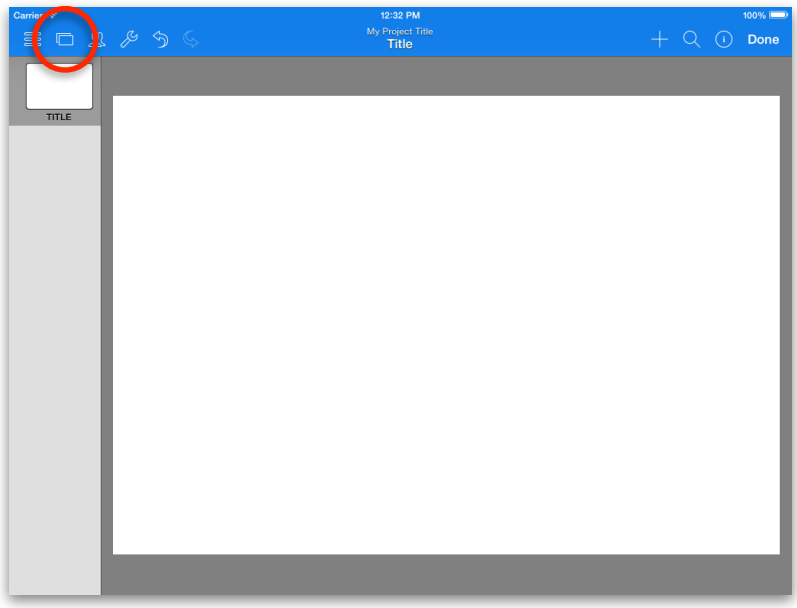
2.3.3 Edit your project

Start by adding a new Page: Enter edit mode and click on the ‘add’ icon. Then click on 'pages' and then New Page.



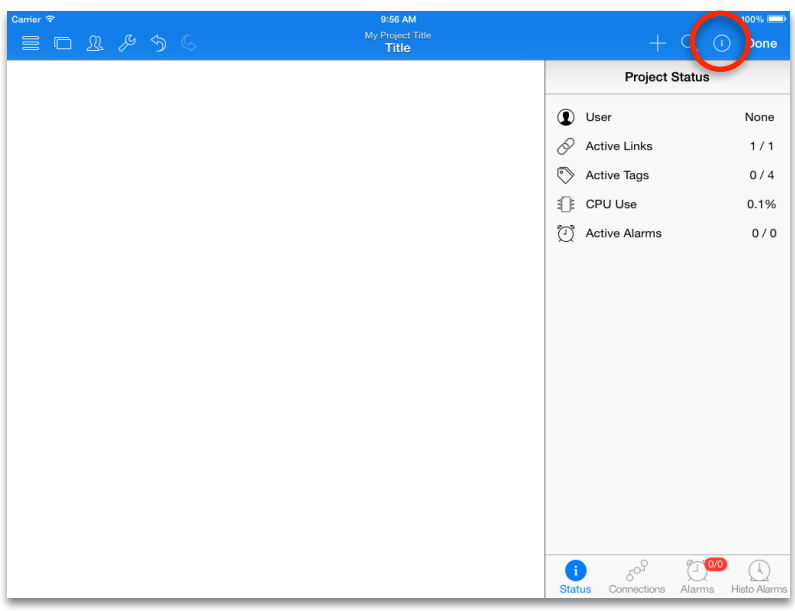


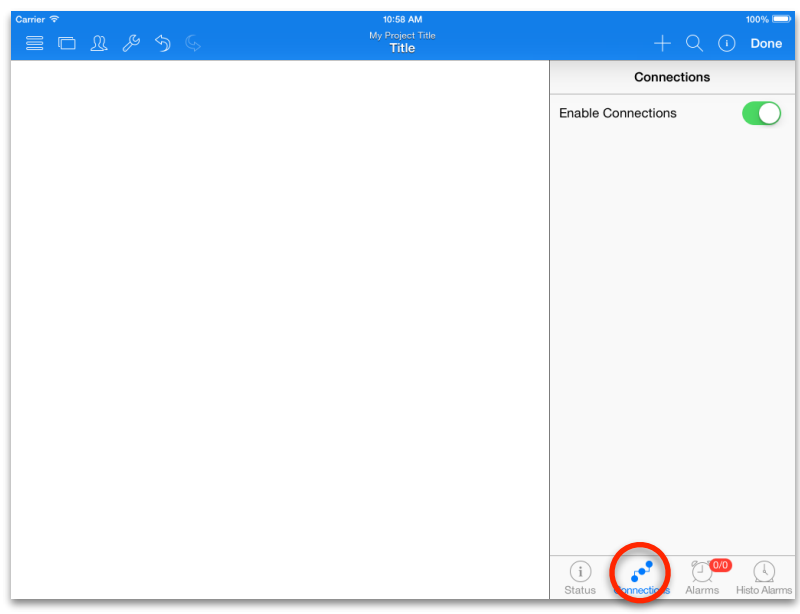
Tap on the 'pages' icon to open the **Page Navigator.** It will appear on the left. To return to full screen mode, click again on the ‘pages’ icon or pan left the Page Navigator



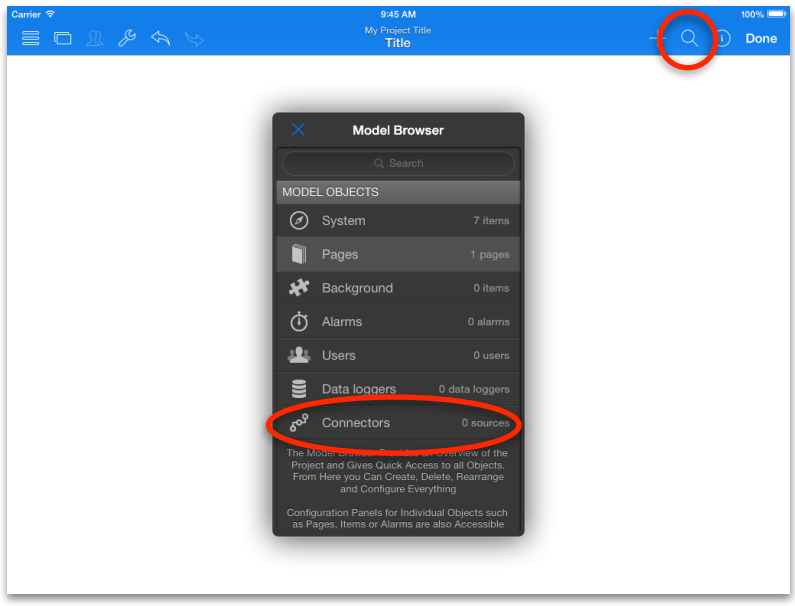
2.3.4 Create a PLC connection

To enable PLC connections, open the **Inspector Panel**, by tapping on the ‘info’ icon, Then click on 'Connections', and switch ‘Enable Connections’ on. By doing this you tell the app to immediately initiate PLC communications as soon as they are available.



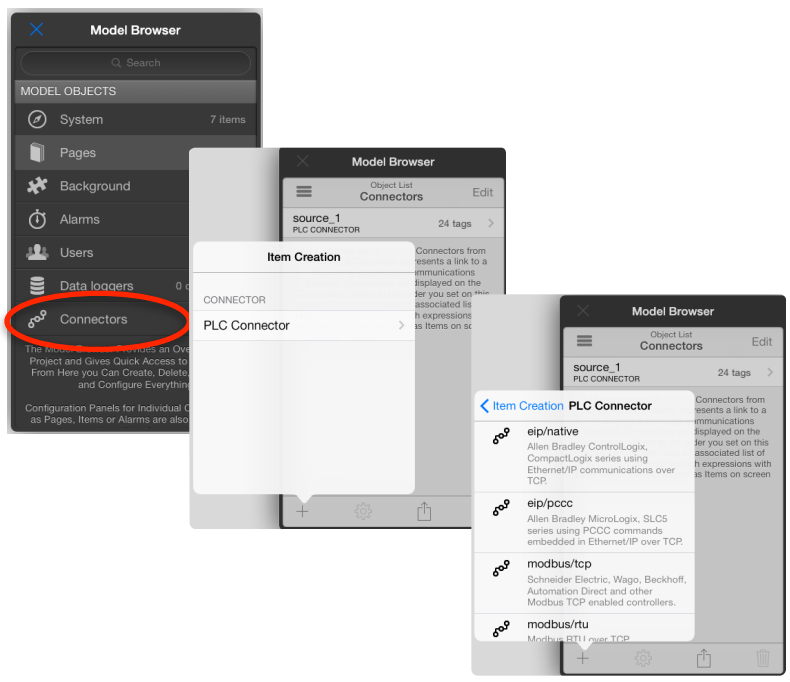


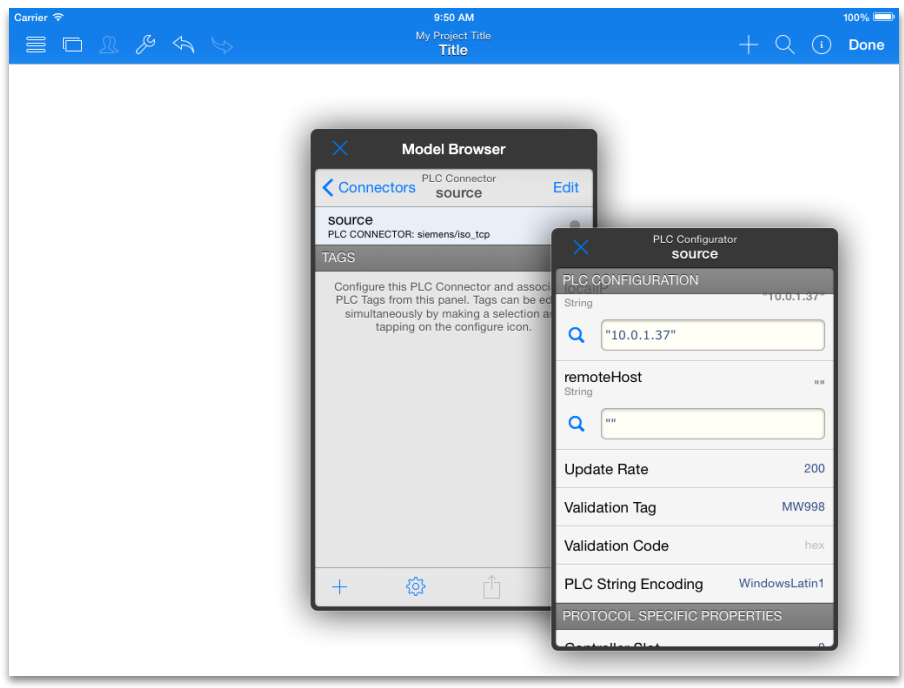
In order to actually communicate with a PLC you need to create a new PLC connector. Tap on the ‘search’ icon to open the Open the **Model** Browser by tapping on the ‘search’ icon. Then Tap on ‘Connectors’.



Tap on the ‘add’ icon to create a new PLC connector. Select one of the available connector types. Your choice will depend on your PLC brand and communication protocol. Once your connector is created configure it by entering your PLC IP address.

Pay attention to the default ‘Validation Tag’ and the ‘Validation Code’. The validation code is zero by default, however each communication protocol defines a default validation tag. You may need to create a validation tag in your PLC if the default one is not available or suitable. Please, refer to the **HMI Draw Reference Manual** for more information on this subject.

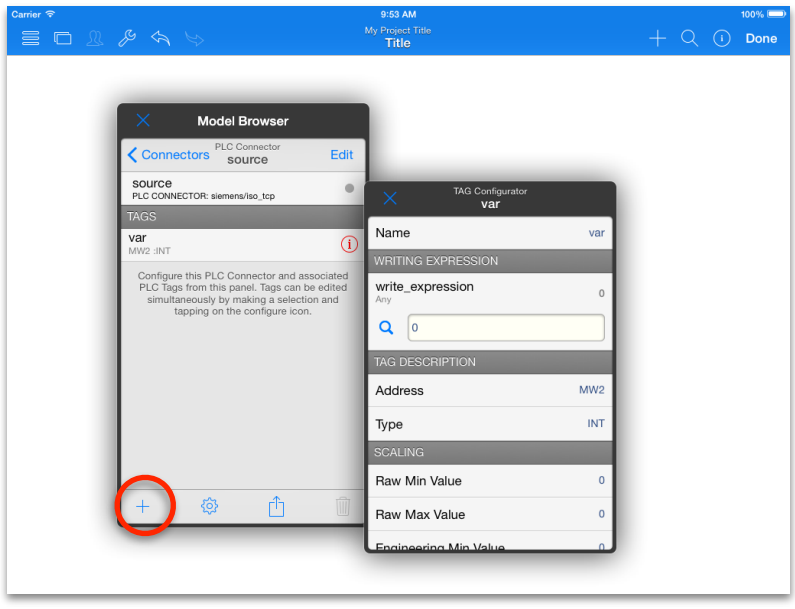


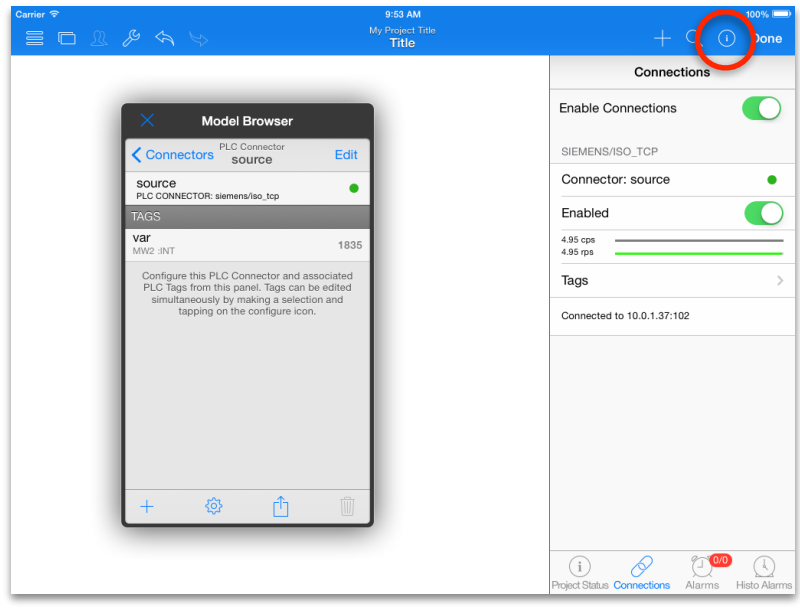


2.3.5 Add tags to your PLC connector

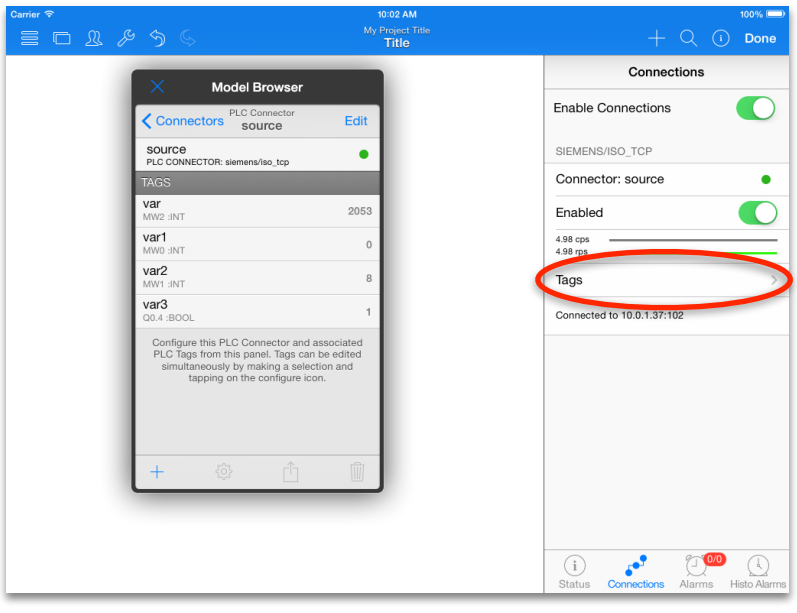
To add Tags, select your PLC connector from the **Model Browser** and tap on the 'add' icon. A tag will be created. Click on the 'gear' icon to open the **Tag Configurator** and enter its 'Address' and 'Type'. Valid entries depend on your PLC communications protocol .They are described in the **HMI Draw Reference Manual**.

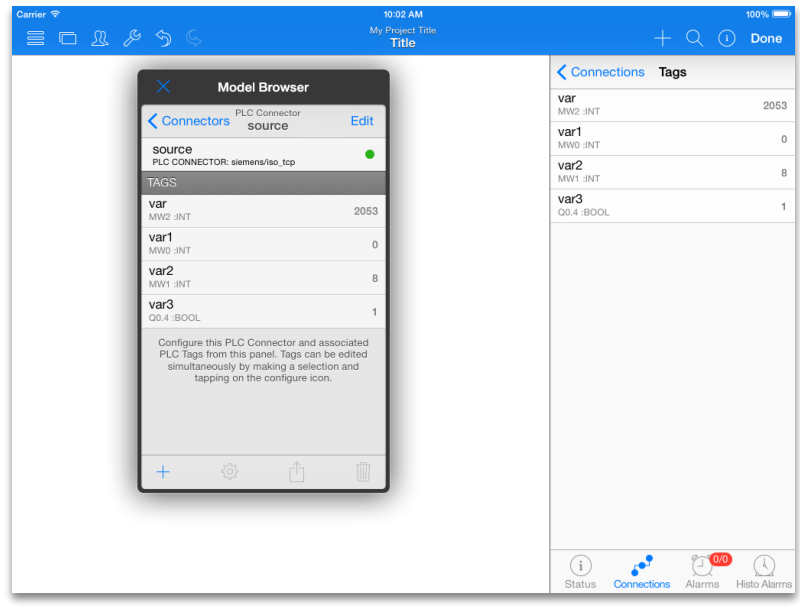
To start communications you must set your PLC connector to enabled on the **Inspector panel.**





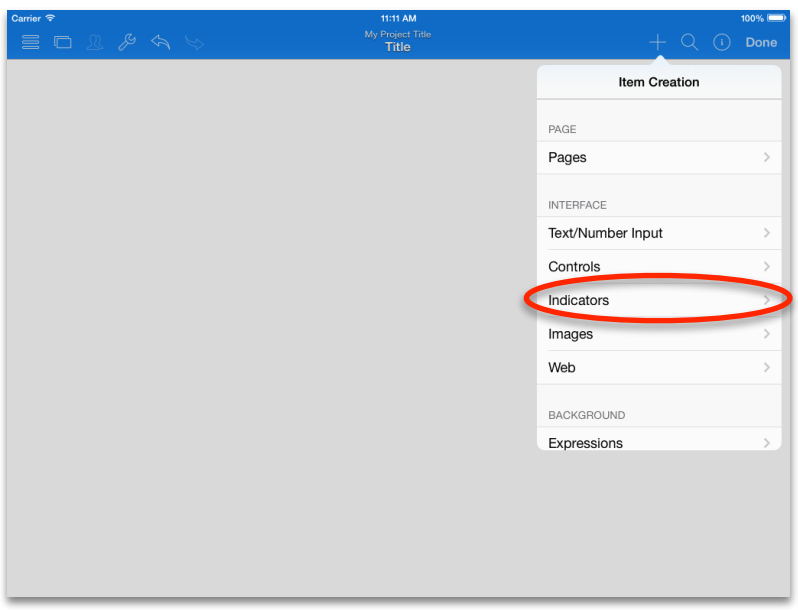
You can see the list of tags for a PLC by tapping on 'Tags' on the **Inspector Panel**. The full list of tags will be presented on the right side of the screen. Observe that tag values are permanently updated on screen as they change in real time.

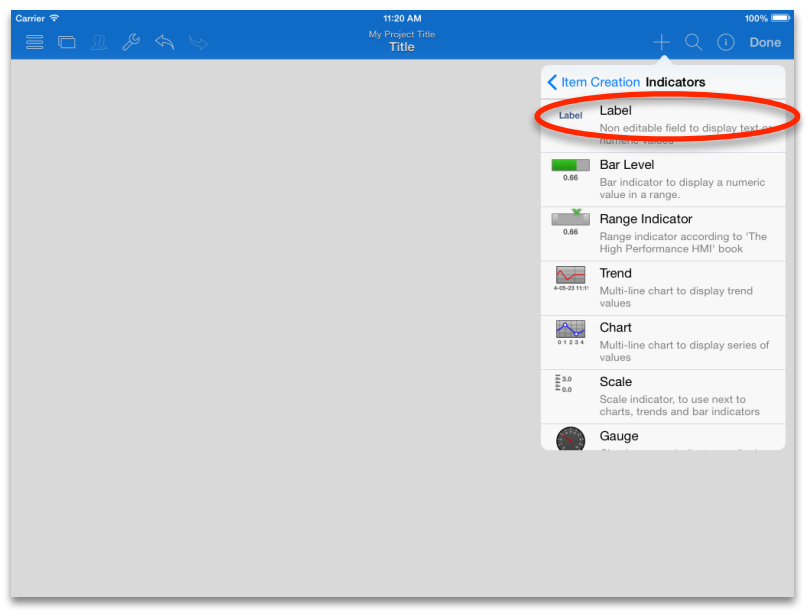




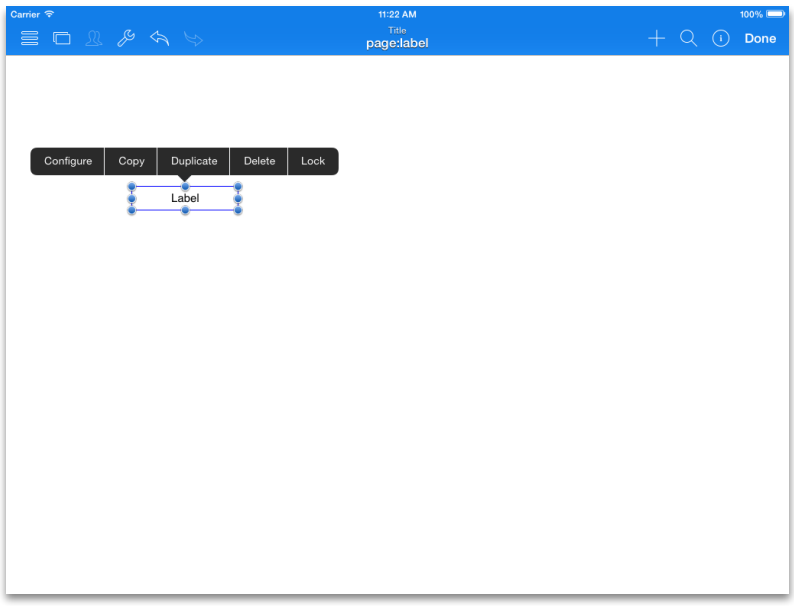
2.3.6 Use an Indicator object to Read a PLC tag value

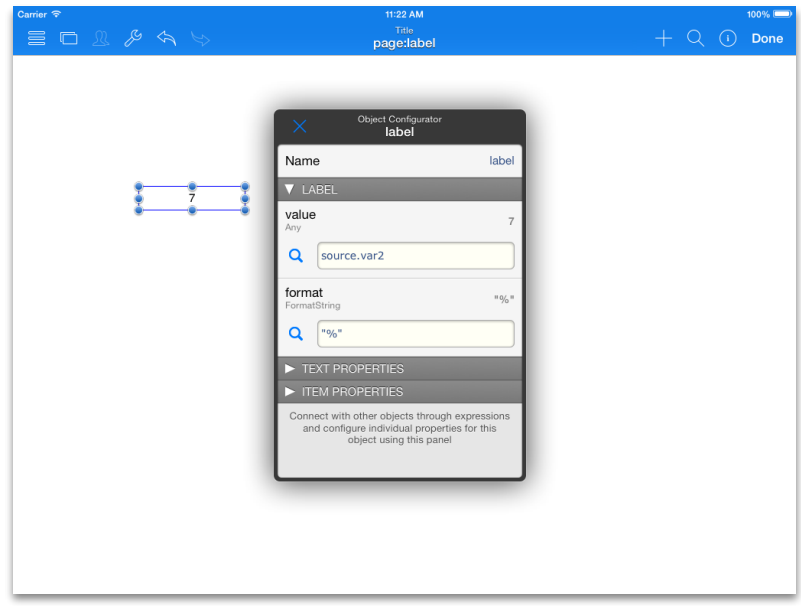
To read tags, try to add a ‘Label.’ object on a page You will find this item on the ‘Indicators’ section.





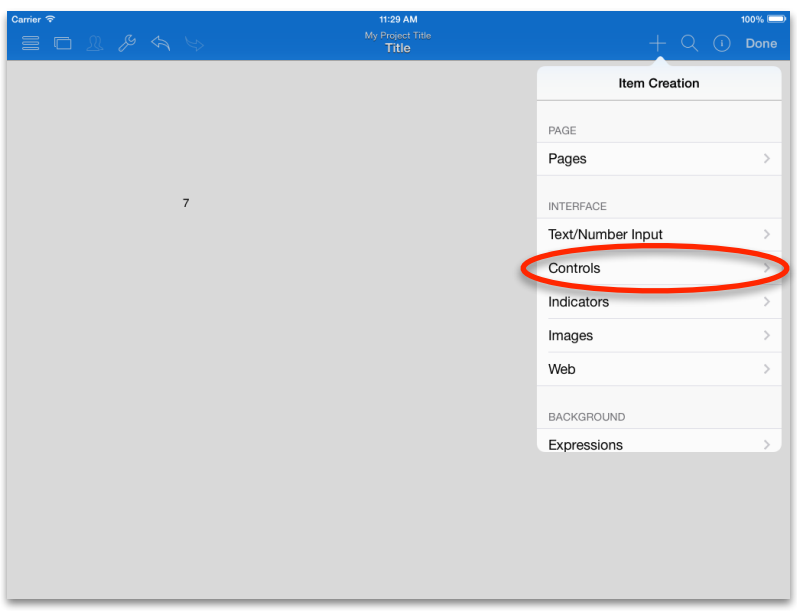
Tap on the label object you just created and press ‘Configure’. The **Object Configurator** for the label will appear. On the 'value' property enter the tag name you want to visualize. For example: *source.var.* The tag value will be shown immediately on the label. If the value changes on the PLC, the label will be updated automatically in real time. You just created a **PLC read link**.

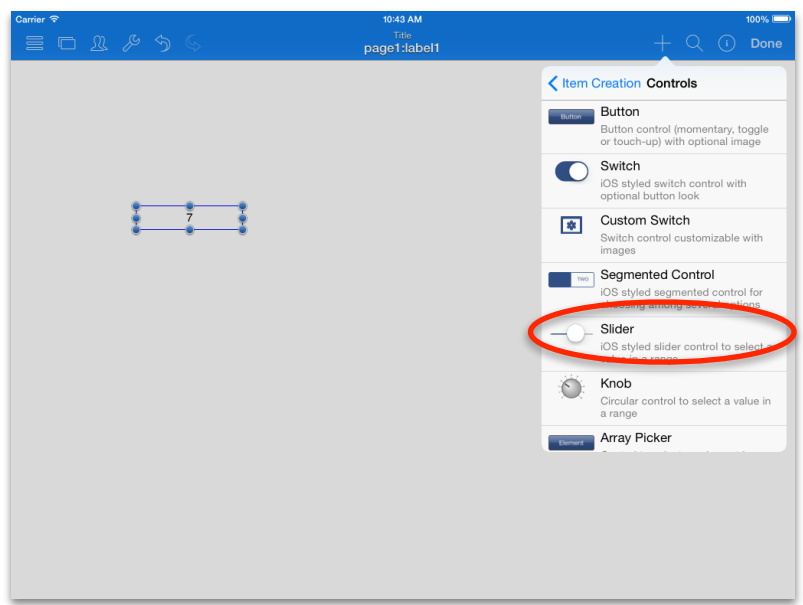




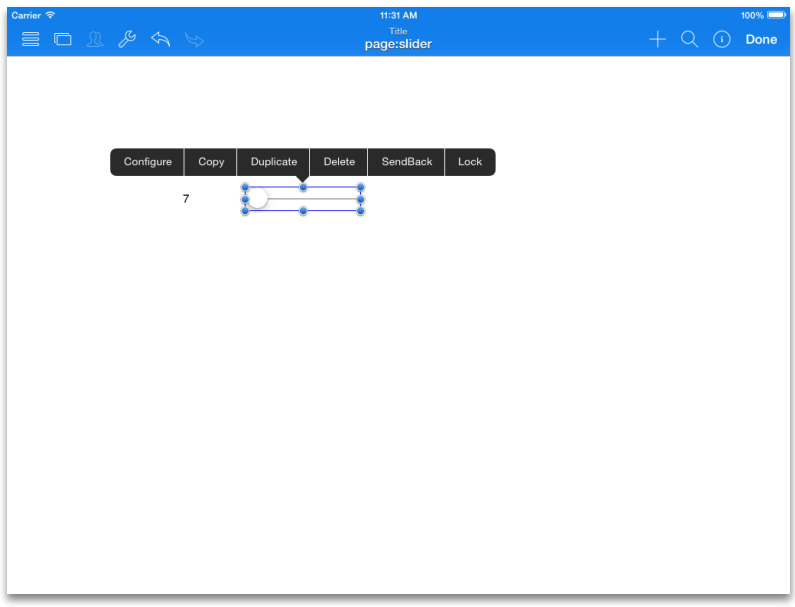
2.3.7 Use a Control object to Read and Write PLC tag values

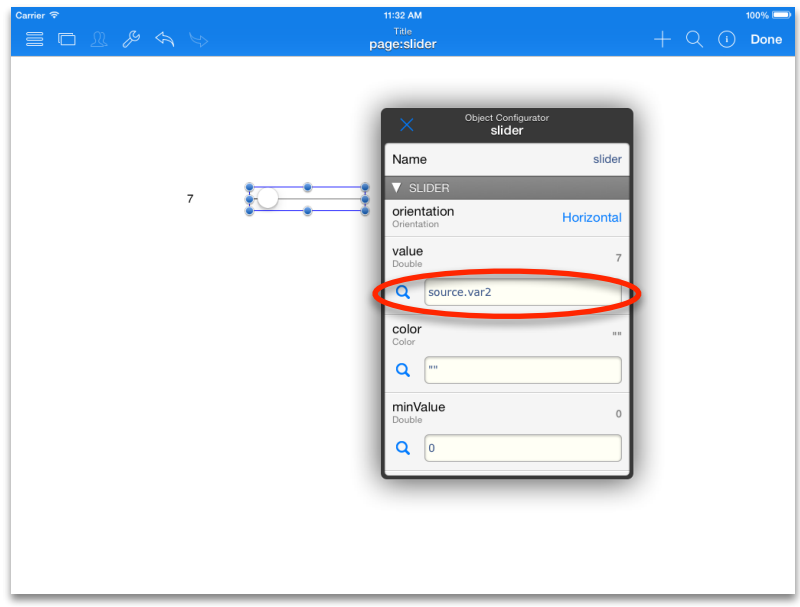
Now add a control item, for example an Slider. Tap on de ‘add’ icon, then ‘Controls’, then Slider, as shown on the screenshot below.





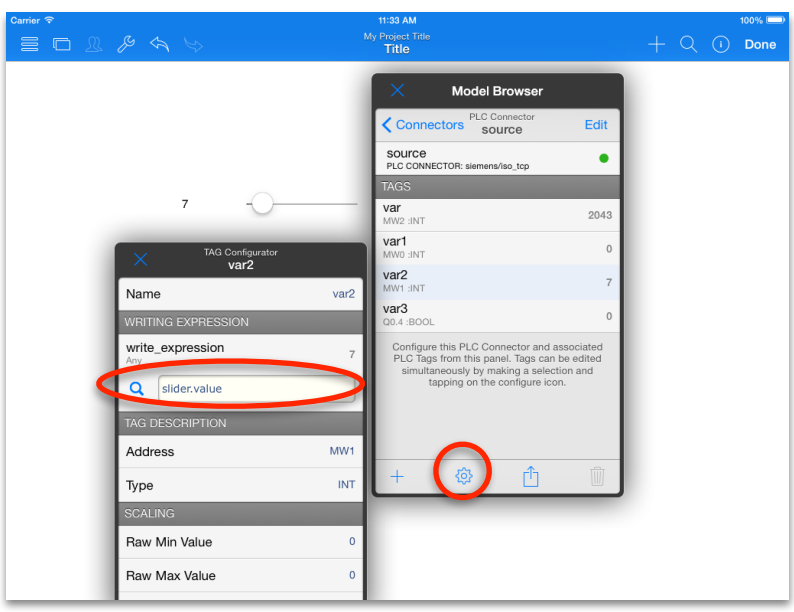
Tap on the ‘Slider’ object and press ‘Configure’. The **Object Configurator** for the slider will appear. On the ‘value’ property enter the name of your tag, for example: `source.var`. The tag value will immediately update the slider position. If the value changes in the PLC, the slider will be automatically updated in real time. You created a **PLC read link.**

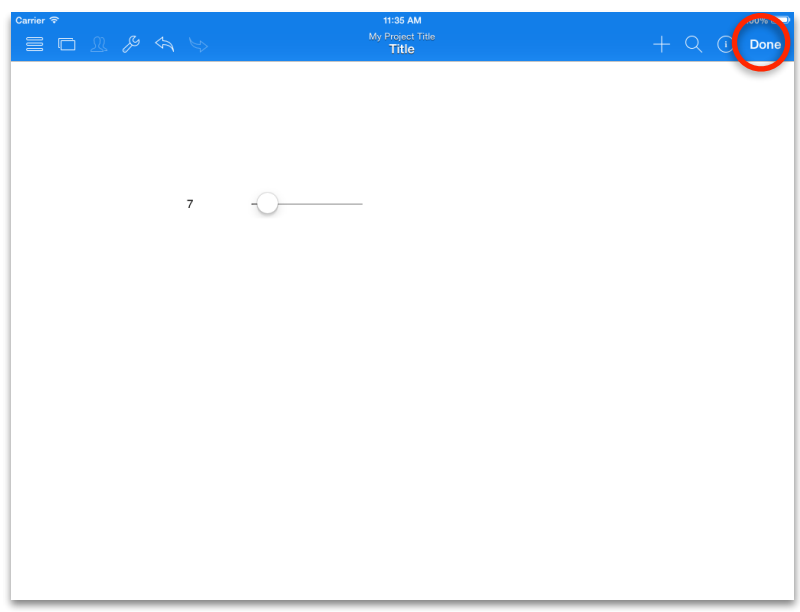




To configure PLC tag writes, open the **Model Browser**. Tap on Connectors, select your PLC connector, select the tag you want to write to and tap on the ‘gear’ icon. This will open the **Tag Configurator** panel for the tag. On the ‘write_expression’ of the PLC tag enter the item property you want to connect for writes. For example you can enter: `slider.value`.

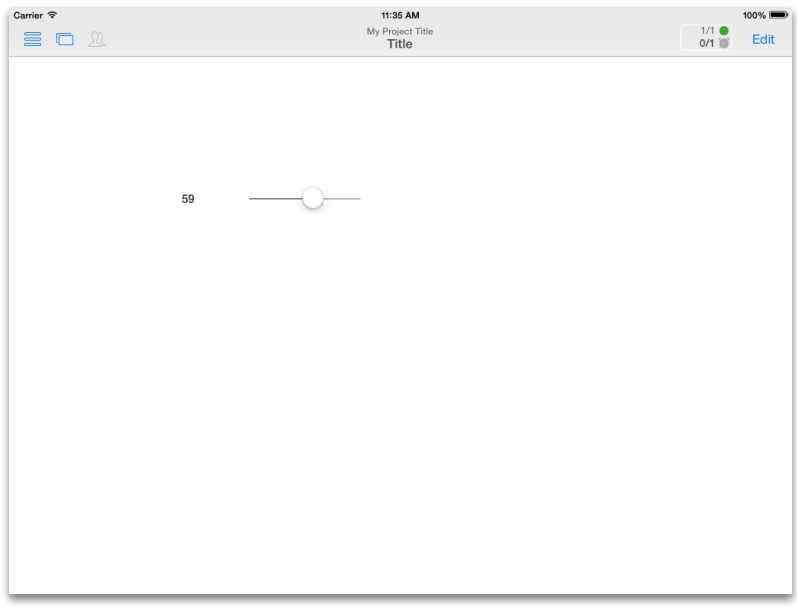
You created a **PLC write link**. Refer to the **HMI Draw Reference Manual** for a detailed description on how this works.





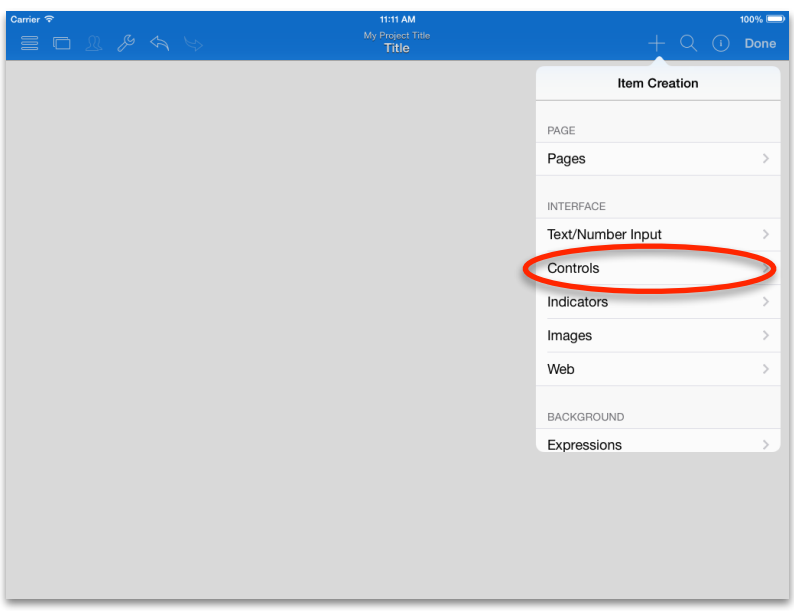
To test your slider read/write functionality, quit edit mode by taping ‘Done’ on the toolbar. Move the slider. After releasing it, the new value will be written to the PLC, consequently the label will update with the refreshed value.

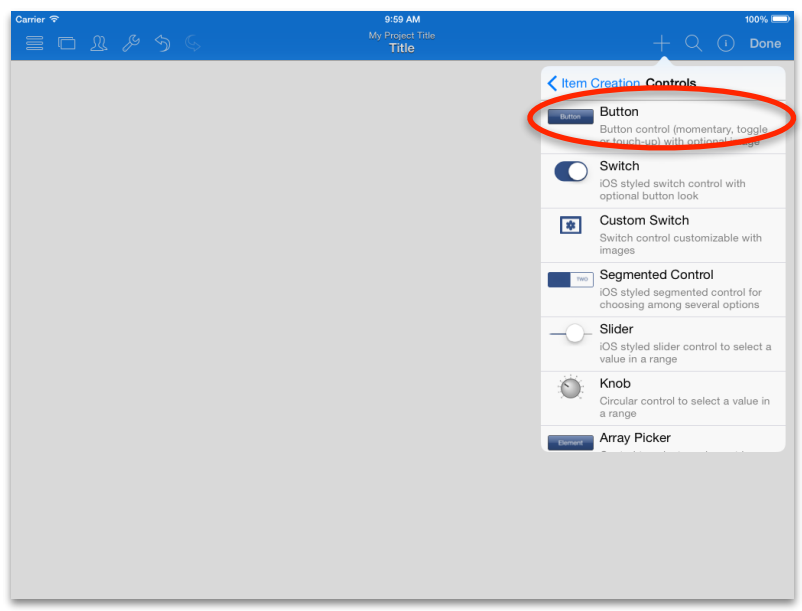
Notice that the toolbar color changed to white when you entered view mode as seen on the screenshot below.



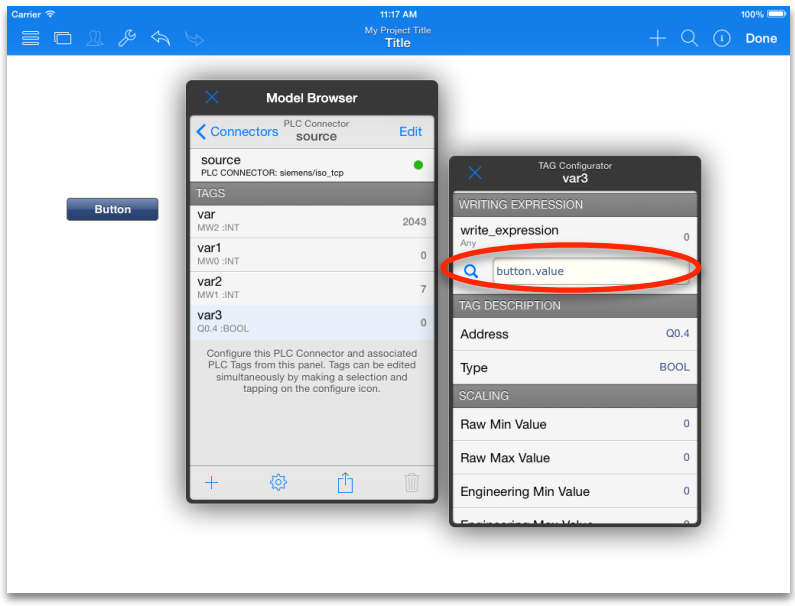
2.3.8 Write to PLC tags without involving reads

Reads and writes are configured independently, so you can chose to enable only writes for a control. The most obvious case is a button. To add a button, enter Edit mode, click on the ‘add’ icon and select the ‘Controls’ category. Tap on ‘Button’.





Now open the **Model Browser** by clicking on the ‘search’ icon. Tap on Connectors, select your PLC connector then the tag you want to connect to and click on the ‘gear’ icon to open the **Tag Configurator**. On the ‘write_expression’, enter the name of the item you want to connect to and its property. For example: `button.value`. You created a **PLC write link.**



To test your button write functionality, quit edit mode by taping ‘Done’ on the toolbar. Press the button object. While the button is down, the Tag value will be 1, after releasing the button, it will switch to 0.

2.3.9 Add custom Texts, Images and Color.

On the previous sections we created *PLC Read Links* and *PLC Write Links* by entering object property names or PLC tag names on objects property fields. However HMI Draw is not limited to this. Indeed, the app supports complex expressions and a number of data types you can use on sophisticated ways to fulfil a variety of needs. All the possibilities are fully covered on the **HMI Draw Reference Manual**

For example you can set the title, the color or the image for a button based on a PLC Tag value such as the running state of a motor. This is all achieved by using expressions. The conditional ternary operator is specially useful for these simple cases.

Assuming you have a tag named *source.motor* indicating the running state of a motor you can make an indicator or control color to change depending on a motor state by entering: `source.motor?"green":"red"` on the 'color' property.

Document Revision History

Refer to this section to look at changes on this document over different versions.

Version 2.2

- Initial Release.

SweetWilliam, S.L. Contact Information

SweetWilliam, S.L.
Science and Technology Park of the University of Girona,
Emili Grahit, 91
(Narcís Monturiol building, P3-B03 office)
17003 - Girona- Spain
Tel: +34 972 18 32 44

e-mail: support@sweetwilliamsl.com
Web: http://www.sweetwilliamsl.com